Development of an Embedded Solar Tracking System with LabVIEW Motion Control

Seung Jin Oh[a], Jun Ho Hyun[a], Won Jong Oh[a], Yeong Min Kim[a], Yoon Joon Lee[a], Wongee Chun[a*]
[a]*Nuclear and Energy Engineering Dept., Jeju National Univ. Ara 1 Dong, Jeju City, Korea, 690-756*
[*]*Corresponding author: wgchun@jejunu.ac.kr*

1. Introduction

Motion control is a sub-field of automation, in which the position and/or velocity of machines are controlled using some type of device such as a hydraulic pump, linear actuator, or an electric motor. The motion control is widely used in the packaging, printing, textile, semiconductor production, and power plants.

National Instruments LabVIEW is a graphical programming language that has its roots in automation control and data acquisition. Its graphical representation, similar to a process flow diagram, was created to provide an intuitive programming environment for scientist and engineers [1].

Crystal River Nuclear Plant engineers developed automated testing system of nuclear plant control modules in an aging nuclear power plant using LabVIEW to improve performance and reliability and reduce cost[2].

In this study, an embedded two-axis solar tracking system was developed using LabVIEW motion control module.

2. Solar Position and Algorithm

In general there are two methods available for solar tracking to identify and follow the position of the sun at any time during a day: one is the optical method and the other is astronomical method. The optical method is called the 'closed loop system'. It uses several feedback sensors such as position sensors of CdS and a comparator that compares the output signals from CdSs on a feedback loop with the desired signals

2.1 Altitude and Azimuth

The algorithm of solar tracker uses the solar altitude (θ_e) and azimuth (θ_a) that are computed at the location of the system or site. The tracker device must be aligned horizontally to implement the altitude and azimuth angles along with the hour angle and declination angles with respect to the celestial equator or plane.

. The solar altitude and azimuth are given by Eq.(1) and (2)[3].

$$\sin\theta_e = \sin\delta\sin\phi + \cos\delta\cos\phi\cos H \qquad (1)$$

$$\sin\theta_a = sign(H)\left|\cos^{-1}\left(\frac{\cos(90-\theta_e)\sin\phi - \sin\delta}{\sin(90-\theta_e)\cos\phi}\right)\right| \qquad (2)$$

The sign function in Equation (2) is equal to +1 if H is positive and is equal to -1 if H is negative.

2.2 Sunrise and Sunset

The solar tracking system must return to the initial rest position after the sun disappears below the horizon, and it must start to track the sun after the sun appears above the horizon.

The sunrise and sunset time are calculated by Eq.(3)[4].

$$T = H + \alpha - (0.06571 \times t) - 6.622 \qquad (3)$$

where, for sunrise,

$t = N + ((6 - lngHour)/24)$

and for sunset,

$t = N + ((18 - lngHour)/24)$

A right ascension (α) is the celestial equivalent of terrestrial longitude and measures an east-west angle along the equator.

2.3 Software by LabVIEW

In this study, LabVIEW is used for developing the algorithm and application program

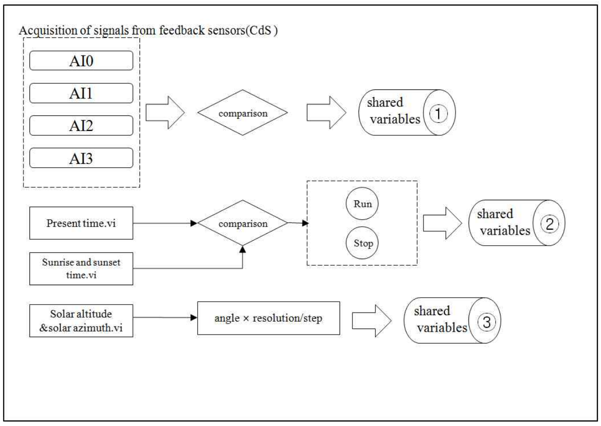

Fig. 1. The structure of the block diagram of input loop.

The algorithm of the developed software is composed of two main loops. Figure 1 shows the input loop of the software by which the solar altitude and azimuth angle and sunrise and sunset time are generated as a shared variable. The present time is compared with the sunrise and sunset time in order to determine whether to run or stop the solar tracker. The AI0~AI3 channels collect the signals from the feedback device, and then the signals are compared with each other in order to confirm that the solar tracker does operate correctly.

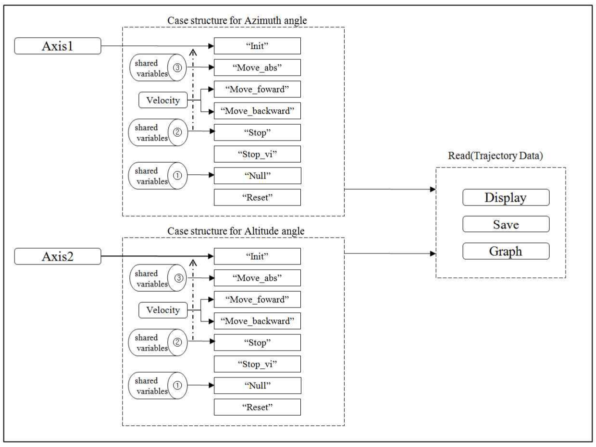

Fig. 2. The structure of the block diagram of move-command loop.

Figure2 shows the part of move-command loop which has two sub-case structures for the azimuth angle and altitude angle.

As soon as the software runs, *present time.vi* returns a time stamp of the current time, which is compared with the time returned from *sunrise and sunset time.vi*. If the current time is the same as the sunrise time, shared variable of run(②) is transmitted to the move-command loop, and then the solar tracker begins to track the sun. If the current time is the same as the sunset time, shared variable of stop(②) is transmitted to the move-command loop, and then the solar tracker stops tracking the sun and returns to its initial position.

The solar altitude and azimuth angle, sunrise and sunset time are calculated in a real time manner at SubVI in the loop showed in Figure 1.

2.4 Analysis of Data

Solar altitude and azimuth angle are calculated by the algorithm written in LabVIEW from which the solar tracker operates in the astronomical mode. These data are compared with those of Naval Observatory which is one of the oldest scientific agencies in the United States, with a primary mission to produce Positioning, Navigation, and Timing (PNT). It is one of the few astronomical observatories and provides a wide range of astronomical data and products.

Figures 3 shows differences of data of altitude and azimuth. As shown in the figure, the differences are less than 0.2º, which are insignificant and do not affect the performance of the solar tracker. Hence, it is confirmed that the algorithm developed so far is reliable and can be used as a servo system by which the tracker follows the sun in an efficient way.

In addition, the sunrise and sunset time of each monthe are calculated , and compared with those times provided from Naval Observatory database. For the sunrise, monthly average difference was less 16seconds and for the sunset, it was less 37 seconds, which does not affect the performance of solar tracking system.

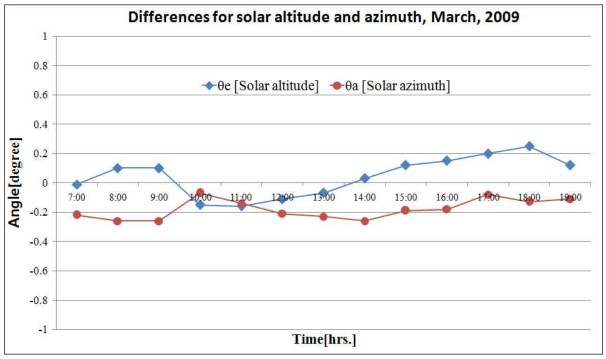

Fig. 3. Differences of solar altitude and azimuth, March, 2009.

3. Conclusions

An embedded solar tracking system was developed for evaluating the performance of LabVIEW motion control.

The tracking algorithm, together with NI-CompactRIO, is applied for the establishment of the stand-alone real time system. Once the developed system is embedded into the system, the system operates in any platform regardless of the operating system.

Being a stand-alone real-time system, the solar tracker returns to its original position at the end of tracking the sun. After sunrise, the system begins to track the sun automatically from its initial position.

A computer program is also developed for control of the solar tracker. The program provides the system with the functions of monitoring the states of system and system error. The solar tracker also implements a measurement in order to generate a database of solar altitude, azimuth, sunrise and sunset and a movement profile such as steps and velocity of motor.

The developed solar tracker with a mini-dish produces high density solar rays, and it could be used for space illumination, irradiation onto the multi-junction cells for electricity production with high efficiency, and for a photocatalyst generation of hydrogen, so on.

REFERENCES

[1]Chance E., Vipin V. et al. National Instruments LabVIEW: a programming environment for laboratory automation and measurement, Journal of the Association for Laboratory Automation. Elsevier, Vol.12, No.1, p.17-24, 2007

[2]Arthur Frink, Automated Testing of Nuclear Plant Control Modules Using LabVIEW, National Instrument, http://200.126.14.82/web/NationalInstruments/Instrupedia/instrupedia/csma/70BCD5798625692D00789D94_1.htm

[3]Duffie JA, Beckman WA. Solar Engineering of Thermal Processes, Wiley, p.13, 2006.

[4]Avitation Formulary V1.44 by ED Willams, http://wolliams.best.vwh.net/avform.